Financial Conduct Authority

Research Note

09/01/2025

A Pilot Study into Bias in Natural Language Processing

Lesley Dwyer, Will Francis, Shalini Tyagi

FCA research notes in financial regulation

The FCA research notes

The FCA is committed to encouraging debate on all aspects of financial regulation and to creating rigorous evidence to support its decision-making. To facilitate this, we publish a series of Research Notes, extending across economics and other disciplines.

The main factor in accepting papers is that they should make substantial contributions to knowledge and understanding of financial regulation. If you want to contribute to this series or comment on these papers, please contact David Stallibrass at David.Stallibrass@fca.org.uk.

Disclaimer

Research notes contribute to the work of the FCA by providing rigorous research results and stimulating debate. While they may not necessarily represent the position of the FCA, they are one source of evidence that the FCA may use while discharging its functions and to inform its views. The FCA endeavours to ensure that research outputs are correct, through checks including independent referee reports, but the nature of such research and choice of research methods is a matter for the authors using their expert judgement. To the extent that research notes contain any errors or omissions, they should be attributed to the individual authors, rather than to the FCA.

Authors

All authors were FCA staff at time of publication.

Acknowledgements

We would like to thank Lawrence Charles, Dan Gibbons, and Fern Watson.

All our publications are available to download from www.fca.org.uk. If you would like to receive this paper in an alternative format, please call 020 7066 9644 or email publications_graphics@fca.org.uk or write to Editorial and Digital Department, Financial Conduct Authority, 12 Endeavour Square, London E20 1JN.

Contents

Non-Technical Summary

Natural language processing (NLP) is one of the fastest growing and evolving areas of data science. Recent breakthroughs in Large Language Models (LLMs) have demonstrated the transformative capabilities of these tools to industry and the public. In the financial sector, firms are exploring ways of integrating language models into customer service chatbots that understand and can respond to queries (Dennehy, 2024). There is interest in the 'robo-advice' field, in which an LLM is used to provide automated financial advice to a consumer (O'Neill, 2023). Firms are also using language models internally to summarise and analyse financial documents with greater efficiency (Lumley, 2023). Innovative use cases like this could help democratise access to financial services and are expected to boost productivity across industries. A McKinsey report suggests that AI technologies such as LLMs could add $13 trillion to global economic output by 2030 (Bughin et al. 2018).

The risks of LLMs have also been widely documented, if not widely understood. However, the still-relevant risks associated with more fundamental NLP techniques such as word embeddings risk being overlooked due to hype over newer LLM technology. Therefore, this research note shares the results of our investigation into bias in word embeddings. Word embeddings are mathematical representations of words that capture what words mean and how they are related. This makes them a useful tool for many NLP applications, and they remain widely used in industry as a cheaper and easier to deploy alternative to LLMs (Goller, 2023). When we conducted a workshop with FCA supervisors, we found a wide variety of potential use cases for word embeddings in financial services. In general, these ranged from consumer-facing chatbots to back-office information retrieval. More recently, they have also become a core feature of Retrieval Augmented Generation (RAG) applications, where they are used to ensure that LLM-generated outputs are contextually correct. However, despite their relative simplicity in comparison to LLMs, word embeddings are still known to encode harmful biases towards demographic groups. These biases could cause tangible harm if embeddings are deployed in consumer-facing applications.

There is a rich literature on this topic of which we were able to cover some of the most fundamental papers. From this literature, we identified then tested a range of techniques for measuring and mitigating biases in six commonly used sets of embeddings. There are three main findings from our work.

First, no individual measurement technique can fully capture bias in embeddings, but using a mix of techniques helps us see bias more clearly. For example, one method, WEAT (Word Embedding Association Test), shows how certain stereotypes, like men being linked to work and women to family, appear in word associations. Another method, Direct Bias, measures how much information about things like gender or ethnicity are encoded in the embeddings. By using both together, we get a better picture of bias.

Second, even when we use multiple methods, tackling bias is still complicated. Existing tools have limits, and bias is often shaped by context, language, and social factors. This means that bias in models needs to be carefully evaluated for each specific use.

Finally, techniques that try to reduce bias, like Hard Debiasing, don't always work as well as we'd hope. While they can lower bias in some areas, they often reduce the overall quality of the model.

While our work does not cover the whole topic of bias in embeddings, it presents a good starting point for any study into this area. Future research could involve testing applications that utilise embeddings, for instance studying the impact of biased embeddings on downstream outcomes for consumers. Mitigation of bias in contextual and sentence embeddings would also be a worthy avenue of enquiry.

1 Overview

Why did we do this research?

In its AI Update (Financial Conduct Authority, 2024), the FCA set out how artificial intelligence fits into its existing regulatory framework and how some of the key elements of that framework will be relevant to firms using AI. The FCA also works with the Bank of England to understand more about how AI is being adopted in industry through the their joint survey on AI and machine learning in financial services.

In particular, as discussed in the AI update (Financial Conduct Authority, 2024), the FCA is enabling a safe and responsible environment for the use of AI in UK financial services. Through an outcomes-based approach, the FCA supports innovation that benefits consumers and markets. In particular under the Consumer Duty framework (Financial Conduct Authority, 2022), firms (in general and through their use of AI) are required to act to deliver good outcomes for their retail customers, in good faith, avoid causing foreseeable harm, and to enable and support retail customers to pursue their financial objectives.

This note presents the results of a technical investigation into biases in word embeddings, which complements other research we are publishing into measuring and mitigating bias in machine learning models.

Given the focus and aims of the work presented readers should be aware that this note:

- constitutes research to spark debate and contribute to academic discussion, rather than any form of guidance or direction about what firms or practitioners should do.
- does not set out any expectations for how firms should approach managing AI risks (in all cases, firms will need to consider the risks relating to AI adoption in the context of their specific use cases and in light of applicable requirements).
- is not a comment or statement about the direction of the broader debates on what constitutes fairness or discrimination in the AI space.

Our interest in the ethics of NLP came from our recognition that machine learning models trained on human-generated text have potential to perpetuate bias and spread discriminatory attitudes and behaviours. A study by Nature (2024) found that AI Chatbots interpret user queries in African American Vernacular English (AAVE) more negatively than queries in Standard American English, causing discriminatory outcomes. Examples like this have abounded since the advent of LLM-powered chatbots.

There is no widely accepted solution for dealing with this issue. Mitigating bias in supervised learning is generally held to be more straightforward due to the presence of labels and well-defined evaluation metrics. In NLP systems, the scale of the training data, the complexity and nuances of language, and the presence of implicit biases make the task much harder.

Word embeddings are a set of mathematical representations of words that reflect their semantic and syntactic attributes. We decided to focus on word embeddings because they are widely used in NLP systems for a range of tasks. Embeddings are also one of the fundamental building blocks of large language models. We undertook this research to identify how biases in embeddings could be identified and removed at source.

In our research we implemented several techniques for measuring biases in embeddings. We also implemented the *Hard Debias* – a technique for measuring and removing the biases.

The goal of our investigation can be summarised by two research questions:

1. Do the bias measurement techniques effectively and comprehensively capture the social biases encoded in word embeddings?
2. Is Hard Debiasing an effective method of removing biases from embeddings?

What did we find in the empirical part of this research?

In this section, we summarise the key findings of the empirical component of our research.

We tested several bias measurement techniques across six open-source embeddings, both before and after debiasing them. We looked at biases relating to six demographic characteristics (which we picked based on what we could measure from the literature):

- gender
- ethnicity
- age
- region
- socioeconomic background, and
- disability

Below we provide a brief outline of our key findings. Overall, in our research, word-analogy tests identify a number of biases in the embeddings. Hard Debiasing does not appear in our work to remove these biases: in some cases, it makes them worse.

Presence of Bias

In our research, the Word Embedding Association Test (WEAT) shows that some pre-specified stereotypes exist in the embeddings, although not to the extent that might be expected. For example, some of the embeddings reinforced the stereotype that age was associated with greater responsibility. Others reflected the stereotype that socioeconomic background is related to education. However, there was no significant evidence supporting the presence of some stereotypes we did expect to see, such as the association of men with work and women with home and family.

The Direct Bias metric showed that the most biased words in the embeddings often aligned with expectations. For example, in one of the embeddings a very male-biased was ‘colonel’ and a very female-biased was ‘ballerina’.

There were cases in our work where the most biased words appeared to be unrelated to the characteristic in question. In these instances, we hypothesise that that Bias Direction

(see p. 13 and p. 23) was poorly defined or the embedding itself was not suitable for use with individual out-of-context words.

Mitigating Bias

In our work, where WEAT did identify the presence of stereotypes, Hard Debiasing was generally reliable at removing them.

Likewise, we found that Hard Debiasing was effective at removing the Direct Bias from word vectors, although this is to be expected and is not a sign that *all* bias has been completely removed.

We find that it is possible to train a classifier to predict the bias association of a word from its debiased vectors. This demonstrates that, in the embeddings we tested, information about the bias is encoded *throughout* a vector and persists even after debiasing.

Key insights from our research

Our research experience has highlighted several important considerations when exploring bias in NLP-based applications:

1. **Assessing Biases in Context**: From our review of the literature, it is consistently stressed that measurement of biases occurs within specific socio-technical contexts. For example, biases present in language embeddings may influence application outputs, particularly when users provide different demographic information or express queries using diverse linguistic styles. Investigating how these biases might surface in consumer-facing applications, like Retrieval-Augmented Generation (RAG) systems, might be able to offer useful insights into their potential effects on fairness and accuracy.
2. **Employing a Range of Metrics**: Since bias metrics are based on specific definitions, we show that no single metric is likely to capture all aspects of bias. In our research, it was beneficial to use a variety of metrics to obtain a more well-rounded view. Throughout our research, we've found it useful to remain transparent about the limitations of these metrics, while staying informed about new developments.
3. **Considering Post-Processing Techniques**: In our research, post-processing methods like debiasing were not always able to fully remove bias without affecting performance. Our findings suggest it could be valuable to also consider the inputs used to build embeddings and to carefully assess whether any observed biases are likely to result in practical harms.

2 Research context

NLP in Financial Markets

It is important to consider the applications that word embeddings might be used in because the nature of the harms that could be caused by biased embeddings, would be determined by the nature of these applications.

Word embeddings find use in a range of NLP tasks, such as information retrieval, text classification, machine translation, recommendation systems, and text generation. These uses could see them deployed in consumer-facing applications. In financial services, popular use cases might centre around customer service and financial advice such as enabling 24/7 financial advice, generating personalised guidance, and identifying consumer vulnerabilities. Firms could also harvest past user conversations to re-train embeddings and language models.

Whilst these use cases could generate benefits for consumers, without deeper consideration they could also be built with underlying embeddings that were biased. For example, an AI customer service agent might leverage word embeddings to understand a user's query and return relevant information to them. Biased embeddings in this scenario could lead to a situation where two consumers receive different outcomes because of the way they express their query.

Similarly, a financial advice system based on biased word embeddings could produce unfair or discriminatory outputs if the system makes biased associations from the information consumers provide about themselves. This might be especially relevant for consumers with vulnerabilities.

Word Embeddings and Biased Associations

Conventions and Terminology

Throughout this paper we use the term 'vector' to refer to the representations of individual words. Meanwhile, we use 'embeddings' to refer to the collection of vectors more broadly. References to the 'embedding space' or 'vector space' refer to the range of possible values a vector could take across all dimensions. When referring to a word we place it in quotation marks ('apple') and when referring to the word's vector we use the letter V to refer to a vector and put the word in subscript as in V_{apple}.

In line with convention, we normalise all vectors to unit length. Pre-normalisation, the magnitude of a vector could be influenced by its frequency in the training data. Normalisation removes this factor while preserving the direction of the vectors.

What are embeddings?

Word embeddings are mathematical representations of words that reflect their semantic and syntactic usage. These representations are called vectors and contain a fixed number

of values. Many data science tasks nowadays require the use of natural language. Since computers don't inherently understand language, word embeddings are a bit like a dictionary that tell the computer what the words means. This makes them useful as the basis of many NLP applications, including LLMs, that benefit from understanding the meaning of words, rather than just the frequency with which they occur in a text.

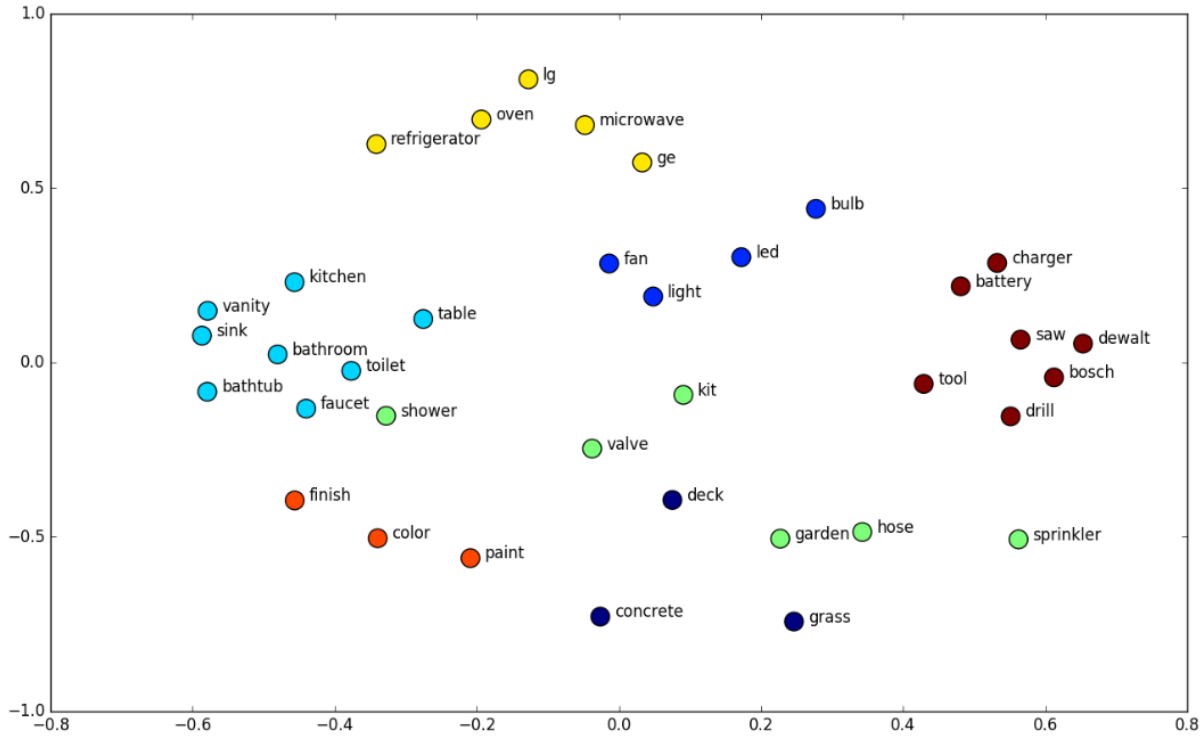
Embeddings are derived from training data: large amounts of real-world text. The vectors are learned by a process that models the relationship between a word and the various contexts it appears within in the training data. The 'contexts' are just the different combinations of words that appear around a word in the training data. For example, in the sentence, 'the cat sat on the mat', we could consider 'sat' to be the target word and 'the cat _ on the mat' to be the context. Across the entire training data, each word will appear in a variety of contexts.

word2vec was one of the first frameworks for learning word vectors and it remains popular. In this framework, the vectors are learned by training a shallow neural network to predict a word from its contexts (or vice versa). If the network is trained to a high degree of accuracy, then the weights connecting the input words to the hidden layer of neurons enable the network to predict the likely context for any given input word. This means that words that appear in similar contexts will be connected to the hidden layer by a similar set of weights, enabling these weights to be taken as the vector representations of the words (Mikolov et al., 2013).

Other methods for generating these representations can be more complicated but always involve modelling the relationship between a word and the contexts within which it appears in the training data. This fundamental intuition means that a word's mathematical vector is a function of its contexts. This results in a very useful property: words that appear in similar contexts will be represented by similar vectors. This property is the basic value proposition of word embeddings: it allows us to measure the similarity of two words by measuring the mathematical similarity of their vectors.

Note that a popular extension of word embeddings involves finding *sentence embeddings* – a vector representation of an entire sentence or paragraph of text. These could be found by simply averaging the word embeddings of the words in the sentence, or with more complicated transformer-based architectures like BERT. While our focus in this note is primarily on word embeddings, much of what we discuss is also relevant for sentence embeddings too.

Figure 1: Visualisation of word embeddings in two dimensions



(Barla, 2024)

Figure 1 shows a two-dimensional visualisation of a set of word embeddings. Words that belong to similar categories (such as bathroom, bathtub) are marked in different colours, and we can see that these vectors tend to cluster together. Since words like 'bathroom' and 'bathtub' appear in similar contexts, they have similar vector representations. We can observe *how* similar two words are by observing how close they are in the vector space. For example, 'battery,' 'charger,' and 'tool' all appear in the same cluster because they have similar vectors. But of the three, 'charger' and 'battery' are the closest, suggesting that these are the most similar words.

The ability to measure the similarity of words makes word embeddings an extremely useful tool in a range of NLP tasks. For example, information retrieval systems might make use of embeddings. An organisation might have a range of internal documents containing different information. An employee or customer who needs a specific piece of information would have to sort through the entire selection to find what they need. However, an information retrieval system might utilise embeddings by creating vector representations for each of the documents in its system. The user's query could then be embedded too, and then used to return the document that has the most similar vector.

Embeddings are also used as inputs for text classification systems. For example, a firm might be interested in predicting the sentiment of its customer calls. Utilising word embeddings would enable the sentiment analysis model to leverage information about how similar words are when making its prediction. This would lead to more accurate sentiment classifications.

Why are embeddings biased?

In the context of language models (as opposed to our other Research Notes which focus on machine learning and pricing), bias as a general term refers to any predisposition in the use of language on the basis of some demographic characteristic. In this paper, we focus on intergroup bias, which refers to situations where word embeddings perpetuate harmful stereotypes or assumptions about social groups. For instance, this would include associating men with the world of work and women with family and caregiving. NLP models are trained on texts written by human authors with those biases or for example with history textbooks are describing a world which was (and is) biased. This process "bakes in" the biases into the models, making them, in a famous phrase, act as "stochastic parrots" (Bender et al. 2021).

When considering *why* embeddings can be biased, it is important to note that word vectors do not strictly reflect word meaning as it is sometimes suggested. It is more accurate to say that a vector reflects the way the word was used in the training data. This is a subtle but important distinction when it comes to thinking about bias. It is fair to say that embeddings encode information about a word's meaning, since a vector reflects the word's usage, and its usage is informed by its meaning. The words 'apple' and 'banana' have similar meanings because they both refer to fruits. Therefore, the two words are likely to appear in somewhat similar contexts and will have somewhat similar representations. But embeddings encode *more* than just the word's meaning because the usage of a word is also informed by the facts, attitudes, opinions, and biases that the word was used to convey across the training data. Word embeddings encode information about all of these *as well as* what the words mean. Data used to train word embeddings has often been harvested from the internet, which is known to contain a deep and diverse array of attitudes, opinions, and biases. Therefore, efforts need to be taken to detect and, where possible, nullify these biases in word embeddings.

There have been a number of publicised cases of real-world NLP systems displaying bias. In 2018, Amazon were forced to scrap an NLP-based recruitment tool it had used to screen the CVs of job applicants after it was found that it was biased against women (Dastin, 2018). The tool had been trained on historic CVs submitted to the company, which disproportionately came from men. Consequently, the tool learned that male-related words were more associated with technology and employment-related words than female-related words were. While this may have been an accurate representation of the data used to train the tool, it clearly constituted a harmful and unacceptable social bias and demonstrates why we need to address bias in these models.

It is important to recognise that biases in word embeddings are harder to address than biases in supervised learning models. In the latter, it is possible to directly observe differences in predictions or errors for different demographic groups. In contrast, in word embeddings biases are encoded in the vector representations themselves and could manifest in the subtle geometric relations between different vectors.

Literature Review

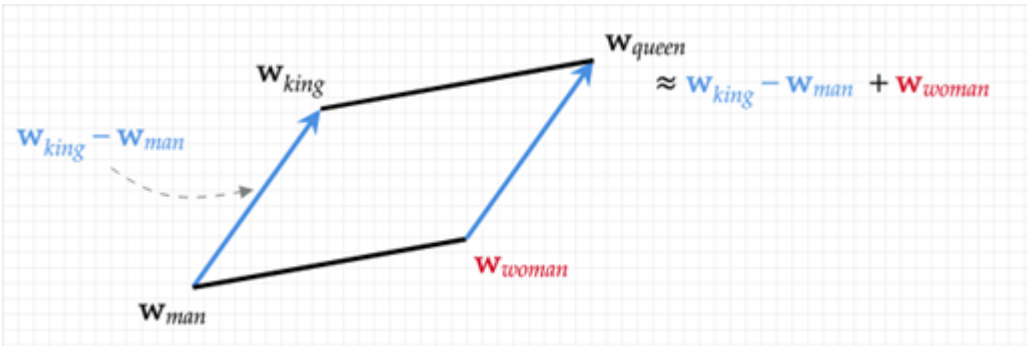
It is this backdrop that motivated our literature review, which aimed to identify techniques for measuring and mitigating biases in word embeddings.

There is an expansive and growing academic literature on bias in NLP. Following the publication of seminal papers that introduced influential frameworks for word embeddings (Mikolov et al. 2013, Pennington et al. 2014), it was soon realised that embeddings picked up and perpetuated the biases present in training data. Consequently, efforts expanded to include the development of techniques for measuring and mitigating these biases. Although a range of such techniques have been proposed, there is ongoing debate over their efficacy and which, if any, are sufficient to fully capture and remove harmful biases. Currently, there is no recognised silver bullet for 'solving' bias in word embeddings or language models.

While our literature review covered some of the better known and more influential papers, there are many more that we didn't get a chance to survey. What we summarise below would be a reasonable starting point for any study of bias in embeddings but should not be taken as an exhaustive list.

A simple technique for measuring biases in word embeddings grew out of a method for assessing the accuracy of embeddings. Mikolov et al. (2013c) suggested that the quality of an embedding could be assessed by testing word-pair analogies. For two word-pairs that express the same semantic or syntactic relation (e.g., man:woman and king:queen, or walk:walking and run:running), a high-quality embedding should encode this pairwise similarity (known as a *linguistic regularity*) in the geometric relations between the vectors for those words. Since the word pairs ['man','woman'] and ['king':'queen'] express the same semantic relation (gender), the geometric relation between the vectors V_{man} and V_{woman} should be the same as the geometric relation between the vectors V_{king} and V_{queen} ($V_{man} - V_{woman} = V_{king} - V_{queen}$). This can be tested with simple vector arithmetic. If the vector offset [$V_{king} - V_{man} + V_{woman}$] is close to V_{queen}, then the embedding has successfully preserved the linguistic regularity and offers the correct answer to the question 'man is to woman as king is to …?' This is shown in Figure 2.

Figure 2: Visualisation of the vector arithmetic for King – Man + Woman = Queen



(University of Edinburgh, 2024)

Since the solution to the vector offset is unlikely to perfectly match the vector for the expected solution, it is typical to use cosine similarity to measure their similarity. Cosine

similarity is measure of similarity between two vectors. The cosine similarity between two vectors *A* and *B* is given by:

$$\frac{A \cdot B}{||A||\,||B||}$$

Where $A \cdot B$ is the dot product of the vectors and ||A|| and ||B|| are their respective magnitudes. Two identical vectors have a cosine similarity of 1, orthogonal vectors have a cosine similarity of 0, and two diametrically opposed vectors have a cosine similarity of -1. For any incomplete analogy in the format "A is to X as B is to …?", the embedding produces an answer by finding the vector that has the highest cosine similarity with the vector offset $[V_X - V_a + V_B]$.

If the embeddings' answer to the analogy constitutes something stereotypical, then we can conclude that the embeddings are biased. Using word2vec embeddings trained on Google News data, Bolukbasi et al. (2016) observe that the answer to the analogy 'man is to computer programmer as woman is to …?' is 'homemaker' ($V_{computer\_programmer} - V_{man} + V_{woman} \approx V_{homemaker}$). This reflects a sexist occupational stereotype and was clearly present in the training data the embeddings were derived from. The authors also find a range of stereotypes relating to ethnicity, and this methodology could easily be extended to other demographic characteristics like disability. An embedding that produces both factual and biased analogies would be problematic because the 'accuracy' of the embeddings would be assured by the former, therefore suggesting that the latter are not down to chance but are due to biases in the training data.

This can be a helpful tool for identifying biases we might expect to see. However, its limitations are that it relies on defining analogies in advance and it only captures biases in the words that constitute the analogy. This makes it a useful heuristic approach for identifying biases in the first instances, but not something that can be relied upon to capture the broader picture of bias across the embedding.

Subsequent papers paid greater attention to developing more robust and comprehensive metrics that captured the broader picture of bias. The best-known example of this is the Word Embedding Association Test (WEAT) (Caliskan et al. 2016). The creators of WEAT argue that word embeddings "necessarily reflect regularities latent in our culture, some of which we know to be prejudiced." The WEAT was inspired by the implicit association test in psychology, which measures the differences in response times when subjects are asked to pair two concepts, they find similar as opposed to two concepts they find different. The metric borrows the intuition that bias can be measured as the differential similarity between sets of words.

In general, two 'target' words like 'man' and 'woman' should be equally similar to two gender-neutral 'attribute' words like 'doctor' or 'nurse'. However, if V_{man} was more similar to V_{doctor} than V_{nurse}, while V_{woman} was more similar to V_{nurse} than V_{doctor}, then the embeddings could be biased. Due to inherent noise in the word vectors, four words alone would not be enough to detect bias. Therefore, the WEAT involves defining large sets of target and attribute words and finding the average differential similarity between the target sets with respect to the attribute sets.

The equation in Figure 3 shows the formally defined test statistic from the original paper.

Figure 3: The WEAT test statistic

The test statistic:

$$s(X, Y, A, B) = \sum_{x \in X} s(x, A, B) - \sum_{y \in Y} s(y, A, B)$$

Where:

$$s(w, A, B) = \text{mean}_{a \in A} \cos(\vec{w}, \vec{a}) - \text{mean}_{b \in B} \cos(\vec{w}, \vec{b})$$

(Caliskan et al., 2016)

By means of example, consider the target sets *X* and *Y* that consist of male words (['man','male','he', …]) and female words (['woman','female','she', …]), while the attribute sets *A* and *B* consist of career-related words (['career','work','employ', …]) and family-related words (['family','home','childcare', …]). For a male word, WEAT finds the average similarity between the male word and all career words, and then subtracts the male word's average similarity with all family words. It repeats this for every male word and finds the sum. It then repeats the entire process for all female words and subtracts the average from the male average. The Null Hypothesis is that the test statistic is zero and there is no significant difference between the two sets of target words in their relative similarity to the two sets of attribute words. The null hypothesis is rejected if the test statistic is not zero and the p-value (derived from a permutation test, where p is the proportion of permutations of male and female terms that produce a more extreme test statistic than the original version of male and female terms) is below a threshold of statistical significance.

In the original paper, the authors find that WEAT scores 'match human biases and stereotypes closely.' However, the test is not a comprehensive measure of bias as it only pertains to the words in the target and attribute sets. It also looks at average similarity across *sets* of words because individual word vectors might be noisy and not fully capture the intended meaning. For example, the word 'man' is often used in a gender-context but is also used to refer to humankind ("mankind's achievements") in general and can be used as a verb ("man the stations"). The vector for this word will reflect *all* of these usages. The authors assume that defining a set of male- and female-related words will see these small differences cancel out and converge on the intended semantic attribute, but there is no reason to think that this would necessarily be the case in practise.

WEAT also requires a pre-determined notion of the biases you're looking for as the sets of attribute and target words must be defined in advance. Unspecific types of biases or biases that aren't captured by these definitions would go undetected. Nonetheless, WEAT remains a useful tool for detecting biases across an embedding, as a more robust measure than the simple analogy test. While it may not exhaustively capture *all* biases across an embedding, it will give a good idea of the *kinds* of biases that are there.

Another popular method involves measuring the *Direct Bias* of a word vector by finding the magnitude of its projection onto a bias 'direction' (Bolukbasi et al. 2016). The bias direction is a 'low dimensional subspace in the embedding that empirically captures much of the bias.' It is found by defining a set of word-pairs that reflect the characteristic of interest. For example, to find the gender direction, you would define gender-definitional word pairs like ['man':'woman', 'he':'she', 'husband':'wife'] and then find the *vector*

difference for each pair ($V_{man} - V_{woman}$, $V_{he} - V_{she}$, $V_{husband} - V_{wife}$). Since the words in each pair are semantically different only by gender, the vector differences should therefore capture the gender information that the vectors encode.

Principal Component Analysis (PCA) is performed on the vector differences and the first principal component is taken as the gender direction, as this is the direction in the embedding that best explains the differences between the gender-definitional terms. The amount of gender information encoded in any vector can be found by measuring the alignment of the vector with the gender direction. Since gender-neutral words like 'doctor' should contain *no* gender information, any alignment with the gender direction constitutes gender bias. Note: if the vectors are normalised to unit magnitude, as is common when working with word embeddings, then the magnitude of one vector's projection onto another is equivalent to the cosine similarity between two vectors.

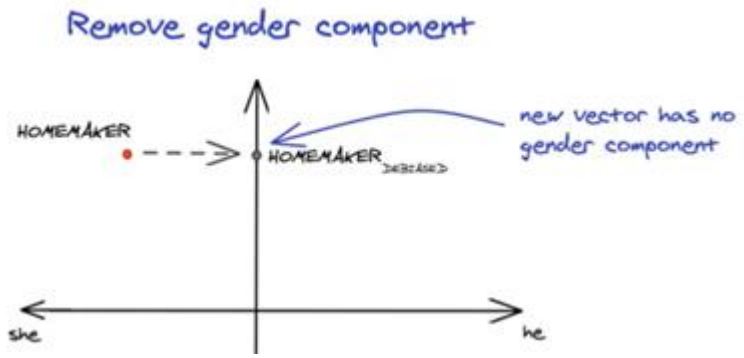
The Direct Bias in any vector is found by measuring its cosine similarity with the bias direction. This enables you to compare the biases encoded in different words and identify the most biased words in each direction. For example, a positive score might indicate a strong male component to the word, while a negative score would indicate a strong female component. Looking at occupations, the authors of the paper find that the words 'homemaker', 'nurse', and 'receptionist' are the most strongly female words, while 'maestro',' skipper',' and 'protégé' are the most strongly male words. A similar process could be repeated to find other biases, such as those related to ethnicity or disability. There are also implementations of this technique that expand the approach to 'multi-class' bias, which makes it suitable for cases such as ethnicity where the characteristic is not easily defined in a binary way.

A major limitation of this technique, however, is that the first principal component of the vector differences needs to have a high explained variance in order to capture most of the information about the bias in the embedding. In the original paper, the first principal component explains around 60% of the variation in the gender difference vectors. However, it is common in practise for the bias direction to explain a much lower amount of variation. In this case, the bias direction isn't useful as it is failing to capture the majority of the bias information. Another major criticism of this approach is that the vector differences themselves might not capture all of the ways that bias manifests across the embedding. We will discuss this more later.

The authors build on this measurement technique by proposing the *hard debias*, a method for removing biases from an embedding. The stated goal of Hard Debiasing is to remove the biased components of word vectors while preserving the useful properties of the embedding. For example, V_{doctor} should be no more similar to V_{man} than V_{woman}, but it should retain its prior similarity to V_{nurse} and $V_{hospital}$. After defining the bias direction with the steps described above, the hard debias involves adjusting the values of the vectors by *neutralising* neutral words and *equalising* definitional pairs with respect to the bias direction. For example, if Hard Debiasing embeddings with respect to the gender direction, neutralisation would consist of projecting every gender-neutral word onto the subspace of the embedding that is orthogonal to the gender direction. This removes gender information from those word vectors by ensuring that their projection onto the gender direction is zero. In figure 4, the word homemaker exhibits a female bias, as it sits further towards the female end of the gender direction (shown on the x-axis).

Neutralising the vector sees it projected onto the orthogonal subspace (represented by the y-axis) so that its gender component is zero and it encodes no gender information.

Figure 4: Removing the gender component



(Mukul Rathi, 2021)

In equalisation, gender-definitional pairs like V_{man}:V_{woman} and V_{he}:V_{she} are centred across the orthogonal subspace so that neutral words are equidistant from both words in the pair. This way, the two words in a pair encode the same amount of gender information but in opposite directions.

Hard Debiasing works by removing Direct Bias across the embedding. Only the gender-definitional words retain any variation along the gender direction, as all other words are neutralised. The authors also suggest that hard debiased embeddings produce fewer biased analogies, while maintaining performance on appropriate analogies. While this appears to be successful, there are a number of limitations to the approach. The first relates to the choice of which words to be neutralised. While a word like 'doctor' is clearly gender-neutral, what about a word like 'beard' or 'Alex'? If we *didn't* want to neutralise 'beard' or 'Alex,' a choice would have to be made about which words are gender-appropriate and which are not. To answer this question, the authors manually label a selection of words as gender-appropriate or not and train a linear classifier on the vectors of those words to predict which words in the vocabulary should be neutralised. The problem with using the vectors as features in a model is that if the vectors are biased (as is assumed if we're bothering to hard debias them), then they cannot be relied upon to produce a fair prediction of which words are gender-appropriate and which are not.

A broader flaw of the hard debias technique derives from the fact that the bias direction defined by the approach described above encodes some *but not all* of the bias information in an embedding. This is for two reasons:

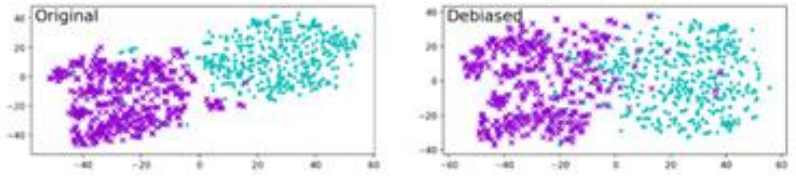
- First, the bias direction is meant to reflect the direction of variation between word vectors in definitional pair. But since it is found using PCA, it only captures this information to the extent that the first principal component has a large explained variance ratio. If the first PC explains less than 50% of the variance, as is common in practise, then the bias direction is a poor reflection of the differences between definitional word vectors.
- Second, the bias direction approach assumes that *all* the information about a characteristic in an embedding can be explained by the vector differences of the definitional word pairs, but this is unlikely to be true in practise. It is likely that information about demographic characteristics is dispersed in more non-linear and

nuanced ways than the bias direction assumes, due to general complexity and subtlety of language.

So even if the first principal component of the vector differences *did* explain 100% of the variance, there is no guarantee that this variance is exhaustive of the information about the characteristic anyway. This flaw in the bias direction would suggest that Direct Bias, which is just a measure of similarity between a vector and the bias direction, is not a complete measure of bias. This would mean that Hard Debiasing, which simply neutralises word vectors with respect to the bias direction, does not completely remove bias. So, while Hard Debiasing reduces bias by the stated definition, bias persists throughout the embedding in more subtle ways.

Gonen and Goldberg (2019) demonstrate this with a couple of experiments. First, they take the word vectors of the 500 most male and 500 most female words (as decreed by Direct Bias) and split them into two clusters using K-Means. Even after the vectors have been debiased, the assigned clusters of the words agree with the initial gender bias with an accuracy of 92.5% (compared to 99.9% when using biased vectors for the clustering). Figure 5 shows the allocation of clusters for the 500 vectors.

Figure 5: K-Means clusters after debiasing align with Direct Bias projections



(a) Clustering for HARD-DEBIASED embedding, before (left hand-side) and after (right hand-side) debiasing.

(Gonen and Goldberg, 2019)

They also train an SVM classifier to predict the Direct Bias gender association of a word vector and find that the model has an accuracy of 88.8% (compared to 98% when the model was trained on the biased vectors). Finally, they define a new measure of bias which looks at the percentage of a word's k-nearest neighbours that share the same stereotypical gender association. They find that this metric has a Pearson correlation of 0.686 with Direct Bias. These experiments demonstrate that gender information is dispersed throughout the embedding, not just the gender direction, and is therefore retrievable even after Hard Debiasing. Even though the vectors have been neutralised with respect to the gender direction, words with the same gender association still have similar values and appear in the same clusters. For example, after Hard Debiasing V_{doctor} might be equally similar to V_{man} and V_{woman}, but it is still more similar to $V_{\text{professor}}$ than V_{teacher}. The problem with the hard debias is that it optimises the embeddings with respect to a metric (Direct Bias) that doesn't fully capture the phenomena it seeks to measure. For this reason, the authors argue, Hard Debiasing is insufficient and shouldn't be trusted as a solution to bias.

The authors level the same criticism at an approach to training bias-free word embeddings from scratch known as *GN-GloVe* (Gender-Neutral Global Vectors). This technique (Zhao et al., 2018) proposes an extension to the traditional GloVe framework by adjusting the loss function during the training process in order to concentrate gender

information in certain dimensions of the vectors. The loss function encourages the vectors of definitional word pairs like 'man':'woman' to vary in one or a small number of dimensions, while the vectors of gender-neutral words are encouraged *not* to vary in those dimensions. These gender-dimensions can then be discarded once the vectors have been learned. The authors showed that GN-GloVe successfully isolates gender information and reduces Direct Bias within the embedding, while still performing well on standard word similarity and analogy tasks compared to baseline methods. However, apart from the resources that would be needed to train embeddings from scratch, Gonen and Goldberg suggest that this approach falls into the same trap as the hard debias. It also conceives of gender bias as the difference between male and female vectors (and encourages these differences to be concentrated in a small number of dimensions). The technique is optimised to reduce bias by this definition but fails to mitigate the broader scope of bias across the embeddings. Like hard debiased embeddings, GN-GloVe embeddings see gender-neutral words appearing in clusters according to their stereotypical gender association. The authors conducted the same experiments and found very similar results (an SVM classifier trained to predict the gender of GN-GloVe vectors did so with 96.53% accuracy).

Gonen and Goldberg's critique of bias mitigation techniques suggest that *any* mitigation effort would struggle: debiasing an embedding will always depend on some definition of bias that it is being reduced or removed. But it will always be difficult to know that this conception of bias is a complete and accurate measure of the biases present in the embedding.

This sentiment was echoed by Caliskan et al. (2022), who developed four simple techniques to demonstrate the subtle and diverse ways that biases can be encoded in GloVe embeddings trained on internet corpora. The authors first look at word frequencies and find that 77% of the 1000 most frequent words in the embedding were more strongly associated with men than women. They then look at the part-of-speech tags of the 10000 most frequent words and find the top male-biased words were verbs, while the top female-biased words were adjectives and adverbs. Next, they conducted k-means clustering of the 1000 most frequent words associated with each gender and interpreted the clusters that emerged in each set. In the male set, clusters corresponded to concepts such as big tech, engineering, sports, and violence. Meanwhile, in the female set, clusters corresponded to concepts like beauty, lifestyle, and cooking. Finally, they looked at the valence (positive/negative sentiment), arousal (activity/passivity) and dominance (control/submissiveness) of words that exhibited male and female biases. Overall, male-biased words tended to be higher on arousal and dominance, while female-biased words were higher on valence. These results demonstrate that biases extend beyond associations between different groups of words: they are also encoded in the types and frequencies of words. Such attributes cannot be mitigated by adjusting the values of vectors.

Our research covered the most fundamental techniques for measuring and mitigating biases in embeddings. Other debiasing methods we didn't survey include adversarial learning (Zhang et al., 2019) and a method for attributing bias to specific training documents (Brunet et al., 2019). We also focused our efforts on identifying techniques for *static* word embeddings, rather than *contextual* word embeddings or *sentence*

embeddings. Nonetheless, our research suggested that there was no single widely accepted method for completely removing biases from word embeddings. Considering this, we wanted to test these techniques with real embeddings to see if we could replicate the authors' results and to test for ourselves if Hard Debiasing works.

Key findings:

The literature suggested there is currently no single effective method for measuring or mitigating bias in embeddings. Compared to the analogy test, WEAT and Direct Bias both provide a more comprehensive picture of bias but have limitations. Similarly, Hard Debiasing might remove some of the bias form an embedding but does not remove all of it, as is shown by Gonen and Goldberg's classification approach.

Having identified these techniques in the literature, our next step was to see how they worked in practise.

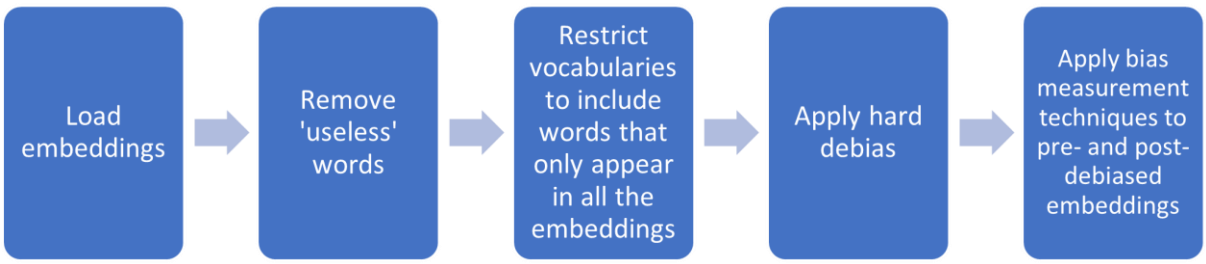
3 Research design

For the empirical phase of this project, we tested a range of the measurement and mitigation techniques identified during the literature review. We used six pre-trained, open-source embeddings to test these techniques. Our analysis pipeline (see Figure 6) involved:

1. Loading the embeddings (see Annex 1, embeddings were chosen due to their availability in open-source, relationship to the characteristics chosen, and suitability for the analysis undertaken here).
2. Filtering out 'useless' words (many of the embeddings featured vectors for short strings of random characters).
3. We did further filtering to restrict the vocabularies to include only the words that appears in *all* of the embeddings. We felt that this was necessary to ensure we could make valid comparisons across the embeddings. This filtering yielded a fixed vocabulary of 58,264 words.
4. We then hard debiased each of the embeddings with respect to six demographic characteristics
5. We applied different measurement techniques on the pre- and post-debiased embeddings to compare the effect of debiasing across the six embeddings.

Our central research question was: does the hard debias actually remove bias from the embeddings?

Figure 6: Our analysis pipeline



The embeddings:

We identified six popular open-source embeddings for our testing, selecting them due to their public availability and potential widespread use in industry. These embeddings are detailed in Table 1. We can split these six embeddings into roughly two categories.

1. Pre-trained vectors. These are embeddings where the vectors have already been learned for a fixed vocabulary of words. They can be downloaded as a file that contains word-vector pairs and then loaded into a dictionary-like object in a Python environment. To distinguish them from the embeddings described below, we refer to these as *static embeddings*. This reflects the idea that the word vectors are pre-trained and fixed.

2. Pre-trained embedding models. These are machine learning models that have been trained to produce a fixed-length vector on demand for any input sequence. They are primarily used to generate *contextual* and/or *sentence* embeddings, but we consider them here anyway. To distinguish them from the *static embeddings* described above, we refer to these as *embedding models*. We installed these models from an open-source repository and then used them to generate vectors for the words that appeared in the static embedding vocabularies. We stored the resulting word-vector pairs in dictionary-like objects.

Table 1: Embeddings

Embedding	N. dimensions	Type	Training data	Available at
GloVe	300	Static embedding	Wikipedia 2014 and Gigaword 5 (6B tokens)	https://nlp.stanford.edu/projects/glove/
word2vec	300	Static embedding	Google News (100b tokens)	https://code.google.com/archive/p/word2vec/
Spacy	300	Static embedding	Curated web data	https://spacy.io/models/en
BERT	768	Embedding model	Wikipedia and BookCorpus	https://huggingface.co/google-bert/bert-base-uncased
SBERT	384	Embedding model	1 billion sentences from multiple datasets	https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2
GPT2[1]	768	Embedding model	WebText (all outbound links from Reddit bar Wikipedia)	https://huggingface.co/openai-community/gpt2

GloVe (Global Vectors for Word Representation) is a widely used word embedding framework developed by Pennington et al. (2014). Here, we use pre-trained GloVe vectors that were trained on the Gigaword dataset and made available by the Stanford team who originally developed the framework. We chose to work with the version with 300 dimensions, as while smaller-dimensional vectors are more computationally efficient, the larger ones provide the most detailed and semantically rich representations. The

[1] We used GPT-2 because this is the most recent open-source version of GPT. While more recent versions may be more accurate, the underlying models not accessible to the public.

developers also made versions available that were trained on Common Crawl and Twitter data. We refer to the GloVe embeddings throughout this research note simply as *GloVe*.

Word2vec is another popular embedding framework developed by Mikolov et al. (2014). The version we use here is a pre-trained word2vec embedding trained on Google News data and made available by the team that developed the framework. We refer to these embeddings as *word2vec*.

The Spacy embeddings were also trained with the word2vec framework but using a wider selection of curated web data. Spacy's embeddings are available as part of the *en_core_web_lg* model that can be downloaded through the Spacy Python package. We refer to these embeddings as *Spacy*.

BERT (Bidirectional Encoder Representations from Transformers) is a popular framework for learning contextual representations of words. The BERT embeddings we used were produced by the bert-base-uncased embedding model that is available on Huggingface. We refer to these vectors as *BERT*.

The sentence-transformers library contains a range of models that have been designed to produce high-quality sentence embeddings. These models are built on top of BERT models by fine-tuning on sentence similarity tasks. Hence, they are sometimes referred to as SBERT. We used the sentence-transformers/all-MiniLM-L6-v2 model from Huggingface to generate vectors for our vocabulary. We refer to this embedding as *SBERT*.

Finally, we used GPT2 to produce vectors for our vocabulary. GPT2 is a language model developed by OpenAI and a predecessor to GPT4. It is available on Huggingface. We used the small version of the model which had 124M parameters.

Debiasing:

We identified six demographic characteristics that we could measure based on existing literature and could be the basis for linguistic discrimination. These were:

- gender
- ethnicity
- age
- disability
- region and
- socioeconomic background

We refer to these hereafter as *demographic characteristics*.

In future work, academics could develop ways to measure bias for a wider variety of demographic characteristics.

We used the *wefe* Python package and its *HardDebias* class to debias the embeddings. This required that we define a set of definitional word pairs (e.g., 'man':'woman', 'he':'she', …) so that we could derive a 'bias direction' for each characteristic. A complete list of the word pairs for each characteristic is available in Table 4 in the appendix. The word pairs that were used were not intended to be definitive or complete – they were simply chosen for the sake of allowing us to test the technique. The word pairs we used also define the demographic characteristics in a binary way, which we did for the sake of

simplicity. However, we recognise that binary definitions can be reductive and that implementations of 'multi-class' debiasing have been developed.

The hard debias framework was designed to debias embeddings with respect to a single characteristic. However, we were interested in the six stated demographic characteristics and wanted to debias the embeddings with respect to all of them. Therefore, we applied the hard debias algorithm sequentially: first we debiased with respect to gender, then we debiased the gender-debiased embeddings with respect to age, then the age-and-gender debiased embeddings with respect to ethnicity, then the same for region, disability, and socioeconomic background. The order in which these debiasing steps are applied may impact the final utility and bias levels of the embeddings, but this is not something we considered in our testing.

Bias measurement techniques:

To determine the impact of debiasing, we applied several bias measurement techniques to the embeddings before and after debiasing. These techniques were:

1. Word Analogy tests
2. Word Embedding Association Test (WEAT)
3. Direct Bias
4. Predicting gender associations with KNN

For the analogy tests we generated a number of incomplete analogies in the format "*characteristic*$_1$ is to *attribute*$_1$ as *characteristic*$_2$ is to _". To solve an analogy, we computed the cosine similarity of each word in the embedding with the vector offset [$V_{\text{attribute1}} - V_{\text{characteristic1}} + V_{\text{characteristic2}}$]. The word with the highest cosine similarity was taken to be the solution. We judged the solution to be biased if it reflected a known or harmful stereotype within the context of the analogy.

To implement the WEAT we again used the *wefe* Python package and its *WEAT* class. We defined ten stereotypes we expected to see across the six demographic characteristics and used *WEAT* to determine if there was statistically significant evidence of the stereotypes being present in the embeddings. We compared the WEAT scores for each stereotype test across the six embeddings before and after debiasing.

To measure the Direct Bias encoded in word vectors we defined bias directions for each characteristic and embedding (6 demographic characteristics x 6 embeddings = 36 bias directions). This was done using the *HardDebias* class from *wefe* (which is a particular implementation of Hard Debiasing) and the same characteristic word pairs we used when Hard Debiasing the embeddings. For each embedding, we calculated the Direct Bias of each word with respect to each of the bias directions. We also calculated the mean and standard deviation of Direct Bias scores across all words in each embedding.

Finally, to predict the gender associations of words from their debiased vectors, we used a K-Nearest Neighbours model with k=5 from *SKLearn*. For each embedding and characteristic, we trained a model on a sample of 1,000 of the most biased vectors and measured the model's accuracy on a test set of 4,000 vectors.

Considerations and limitations:

We note that the word pairs used to define the bias directions may be imperfect due to varying patterns of language usage online. For example, one of our demographic

characteristics of interest was region – we were interested to know if the embeddings reflected regional biases in the UK. We used words like 'north' and 'south' to define this characteristic. However, since embeddings are typically trained on *global* text data, these words will very likely have picked up associations that aren't specific to the UK. Beyond this specific example, we can't be sure more generally that the words we used to define the demographic characteristics were used in the way we were expecting, and so we can't be sure that they are accurately capturing the information we expect.

Related to this, the effectiveness of the bias direction in capturing information about the relevant characteristic depends on the consistency of the vector differences across the definitional word pairs used to define the bias direction. For example, if the vector differences $[V_{man} - V_{woman}]$ and $[V_{he} - V_{she}]$ are highly similar, it suggests that gender information is dispersed consistently throughout the embedding and that the gender direction will be a good approximation of this. However, if the vector differences are noisy and *don't* exhibit strong similarity, then this implies that gender information is *not* uniformly distributed throughout the embedding and that the gender direction will not accurately capture gender information. Since the bias direction is equivalent to the first principal component of the vector differences, its explained variance will reveal the extent to which it captures the flow of information about a characteristic between word pairs. A large explained variance ratio would suggest that the vector differences *are* consistent and that the bias direction is a good approximation of the flow of information between definitional words. Conversely, a low explained variance ratio could suggest that the vector differences are *not* consistent and that information about the characteristic is not distributed consistently throughout the embedding.

Finally, we note that the three embedding models were designed to generate contextual word vectors or sentence vectors, rather than vectors for single out-of-context words.

4 Results

Results summary:

Overall, we found that debiasing has mixed results. In many cases, debiasing increased the number of both incorrect and biased analogies produced in the analogy test. It had some success in reducing WEAT test scores, although this only indicates a meaningful reduction in bias subject to the concepts in the test being well defined. While Hard Debiasing largely reduced the magnitude of Direct Bias across the embeddings, the process is designed to do this, and it should not be taken as a sign that the bias has been completely removed. This is demonstrated by the fact that a binary classifier trained to predict the prior bias association of a debiased word-vector can have better-then-random accuracy (that's to say, better than a classifier which just flips coins with a probability of picking 'prior bias' equal to the proportion of biased words).

Word analogy test:

After Hard Debiasing our six embeddings with respect to the six demographic characteristics, we first looked at the word analogies task. We defined a list of demographic characteristic word-pairs like ['man', 'woman'] and a list of attribute words that included qualities like 'senior' and finance terms like 'mortgage.' The full set of the characteristic-pairs and attribute words is available in the appendix. We used these terms to generate 95 incomplete analogies. For each analogy, we used the approach described in the research design section to find each embedding's solution.

We found that all of the embeddings generated a number of biased analogies. Table 2 shows the number of biased analogies produced by each embedding before and after debiasing.

Before debiasing, the *static embeddings* (*GloVe*, *word2vec*, *Spacy*) were more biased than the *embedding models* (*BERT*, *SBERT*, *GPT2*). This could be because the more advanced training architectures of the *embedding models* were less susceptible to spurious relationships between words in the training data, compared to the simple methodologies that were used to generate the static embeddings.

However, the *embedding models* also produced fewer accurate solutions in general (the exception to this was *SBERT*, which produced fewer biased analogies *and* fewer inaccurate analogies). We judged a solution to be inaccurate if it did not make sense within the context of the analogy (e.g., *GloVe*: "*man* is to *educated* as *woman* is to *elected*").

Table 5 in the appendix shows the number of incorrect solutions produced by each embedding before and after debiasing. The propensity to produce incorrect solutions of embedding models implies that, despite this being their purpose, they struggled to encode the associations between individual words that appeared in the training data. If this is the case, this would suggest that these embedding models produced fewer biased analogies because they failed to encode the biased associations that were present in the training data. Meanwhile, the *static embeddings*, which are explicitly designed for

individual out-of-context words, were better able to encode the associations that exist in the training data, including the biased ones.

Of the six demographic characteristics, disability had the highest number of biased analogies (e.g., *GloVe*: '*able* is to *finance* as *disabled* is to *welfare*'). Following this, socioeconomic background and gender had a high number of biased analogies, while age and region had a relatively low number. Table 3 shows the number of biased analogies relating to each characteristic.

Many of the biased analogies were biased across many of the embeddings. For instance, 68% of the analogies including the word 'senior' were biased. This suggests that that there might be certain associations across training corpuses that all embeddings and frameworks are susceptible to. Similarly, the words representing social attributes or qualities (like 'senior' or 'educated') produced more biased analogies than the finance terms like ('invest'). This could suggest that finance-related language in online text corpuses exhibits less bias in general than language about social qualities.

Table 2: Biased analogies produced by each embedding before and after debiasing

Embedding	Number of biased analogies before debasing	Number of biased analogies after debiasing
GloVe	17	33
word2vec	19	30
Spacy	24	37
SBERT	4	29
BERT	16	41
GPT2	5	27

Table 3: Biased analogies relating to each characteristic before and after debiasing

Characteristic	Number of biased analogies before debasing	Number of biased analogies after debiasing
Gender	15	14
Ethnicity	10	32
Age	7	10
Disability	28	78
Region	8	17
Socioeconomic background	19	54

Interestingly, debiasing actually increased the number of biased analogies produced by all embeddings and for all demographic characteristics except gender. *BERT* and *Spacy* remained the most biased, producing 41 and 37 biased analogies respectively, while *SBERT* remained the least biased, producing 29 biased analogies. After debiasing, disability continued to have the most biased analogies, with 78 in total. There were also significant increases in the biased analogies relating to ethnicity, region, and socioeconomic background.

The number of incorrect analogies also increased after debiasing, suggesting that debiasing damages the overall accuracy and utility of the embeddings. *BERT* and *GPT2*, in particular, produced only a handful of correct analogies after debiasing, while for all embeddings except *SBERT* the majority of the analogies were incorrect.

Many of the models exhibited convergence on a single word for multiple analogies after debiasing. In many cases, this 'convergence word' rendered the analogy incorrect. For example, debiased *GPT2* answered 'uncle' to every analogy involving the terms 'man' and 'woman' (e.g., *man* is to *health* as *woman* is to *uncle*). However, in some cases the convergence word caused the analogy to be biased. For example, debiased *Spacy* answered 'unemployed' to ten of the 15 analogies involving the terms 'rich' and 'poor' (e.g., *rich* is to *invest* as *poor* is to *unemployed*). Examples like this explain some of the increase in the number of biased analogies after debiasing.

These results seem to indicate a clear failure of debiasing, in terms of not reducing the number of biased word analogies. Not only did bias appear to increase, but the overall accuracy of the embeddings seemed to decrease. While the more robust measurement methods might be a better indication of the change in the level of bias in the embeddings, the question over their accuracy after debiasing is worrying.

Word Embedding Association Test:

Next, we tested the Word Embedding Association Test (WEAT). Like the analogy test, WEAT measures the extent to which stereotypes and group associations are encoded within the embedding. It is more robust than the analogy test, though, as it involves measuring the similarity between *concepts* by finding the average similarity between each pairwise combination of the words that constitute the concepts.

For example, we wanted to determine if the embeddings associated *men* more with *work* and *women* with *family*. We defined *men* and *women* (the target sets) as ['man', 'he', 'male', …] & ['woman', 'she', 'female', …]. We then defined two attribute sets representing *work* and *family*, which consisted of ['work, 'job', 'salary', …] & ['home, 'family', 'children', …].

The WEAT score captures the differential similarity of the words that constitute the concepts *men* and *women* with respect to the words that constitute the concepts *work* and *family*. A complete list of the target sets and attribute sets can be found in Table 6 in the appendix. We hypothesised ten stereotypes we expected to see across the six demographic characteristics and tested these for each of the six embeddings before and after debiasing. We compared results with the test effect size, which is the normalised version of the WEAT metric and is more suitable for comparison across target/attribute sets and across embeddings, as it is not dependent on the number of words in each set.

P-values were calculated using a permutation test and any effect size whose p-value was statistically significant at the 5% level was judged to be evidence that the stereotype was present in the embedding.

The results suggested that the stereotypes we hypothesised were not widely present across all the embeddings and were mostly nulled by debiasing where they did occur. In particular:

- Age was found to be strongly biased with respect to responsibility by *word2vec* and *SBERT* and weakly biased by *GPT2*. This means that these embeddings associated the concept of *age* with *responsibility* and the concept of *youth* with *a lack of responsibility*. The strength of the bias decreased slightly for *word2vec* after debiasing but actually increased for *SBERT* and *GPT2*.
- Both *GloVe* and *word2vec* found that *ethnicity* was weakly biased with respect to *risk*, but in both cases debiasing reduced this bias to near 0. *GloVe*, *word2vec*, and *SBERT* all found that *socioeconomic background* was highly biased with respect to *education*, and in all of these cases debiasing significantly reduced the bias.
- Finally, all embeddings except *GPT2* found that *socioeconomic background* was highly biased with respect to *wealth*, and debiasing moderately reduced this for all embeddings. That could of course reflect the fact that one's current wealth might be associated with their socioeconomic background, but if these embeddings are implemented in other language models this could still lead to harmful stereotyping.

Stereotypes that were not found to be significant in any of the embeddings were:

- *age* with respect to *health*
- *disability* with respect to *health*
- *disability* with respect to *risk*
- *gender* with respect to *seniority*
- *gender* with respect to *work/family* and
- *region* with respect to *income*.

These results were interesting as they contradict the authors' findings who introduced this bias measurement technique. In the original WEAT paper, the authors found that in the same word2vec embeddings we used, gender was highly biased with respect to work/family (male terms were more associated with work terms while female terms were more associated with home and family terms). We suspect that our testing did not replicate these findings because we defined the concepts differently. While the WEAT authors defined *male* and *female* with male and female names, we defined these concepts with gender definitional words like 'he'-'she' and 'man'-'woman'. The fact that different ways of defining concepts can lead to different results should be considered a flaw in the WEAT approach, especially since there is no 'objective' way of defining a concept.

Many tests showed results that implied the expected stereotype was there but wasn't statistically significant at the 5% level. Table 7 in the appendix show the effect size and p-value for all tests and embeddings. The results suggest that stereotypes may not have been present in the embeddings to the expected extent, and that debiasing was generally reliable at removing them when they did appear by stripping the target words of their differential associations with the attribute words.

Direct Bias:

WEAT provides an estimate of how biased specific words are in relation to other words. This makes it a useful test for identifying stereotypes and negative associations that align with pre-specified hypotheses. But it does not give an indication of how vectors across the embedding encode biases more generally.

To understand this, we turned to Direct Bias. Direct Bias quantifies the amount of information about a characteristic that is encoded in a word vector. If a word is semantically neutral with respect to the characteristic in question, then its vector should encode *no* information about the characteristic. Therefore, any information that *is* encoded in its vector constitutes bias. For each of the six embeddings, we measured the Direct Bias of each word with respect to each of the six demographic characteristics. This enabled us to compare how bias was encoded across the embeddings and observe the impact of debiasing.

Our findings showed that the most biased words often aligned with known stereotypes. For example, 'colonel' was the second-most male-biased word in the *GloVe* embeddings, while ballerina was the eighth-most female-biased word. Similarly, words like 'physicality' and 'athleticism' were among *GloVe's* most youth-biased words, while 'died' and 'funeral' appeared among the most age-biased. Where results like this appear (see Table 8 in the appendix for the full list of the most biased words for each characteristic and embedding), it suggests that social biases *are* encoded in the embeddings.

However, there were also cases where the most biased words did *not* align with known stereotypes. While the most disability-biased words in each direction in the *SBERT* embeddings included 'stronger' and 'abilities', and 'disabilities' and 'wheelchairs' respectively, this was not the case for *GPT2* embeddings. In that case, the equivalent for the *GPT2* embeddings were 'quartered' and 'angering', and 'excavators' and 'excited', where it is not especially clear at face value which set of words correspond to which side of the bias. The *GPT2* and *BERT* embedding models tended to produce results like this – almost all of their most-biased words were semantically unrelated to the demographic characteristics they were supposedly aligned with. The regional bias tests also yielded odd results across all the embeddings. These examples might suggest that there were no biases encoded in the embeddings and that the most biased words are simply those that align with the bias direction due to randomness in the vectors.

However, it is more likely that these examples are an indication of the bias direction failing to properly capture information about the characteristic. In the case of *BERT* and *GPT2*, this could be because the models are optimised to produce contextual word vectors, rather than static vectors. In the case of the regional bias tests, it could be that the word pairs used to define the region direction were noisy and did not exhibit consistent vector differences. In both cases, the results suggest that the bias direction is not a suitable method for capturing and measuring biases.

The average bias across all words gives a sense of how far biases are encoded across the embeddings. Figures 7 and 8 show the mean and standard deviation of Direct Bias in each embedding for each characteristic before and after debiasing. If the embeddings contained no biases, you would expect to see both metrics close to zero. *word2vec* had mean values close to zero for all demographic characteristics but standard deviations

between 5.8 and 7.6, suggesting that there was a large degree of variability between words.

Meanwhile, *GloVe* had larger means in absolute terms *and* larger standard deviations, suggesting that this embedding encoded more bias than *word2vec* overall. *SBERT* had mean values in a similar range to *word2vec* and *GloVe* but typically lower standard deviations, suggesting lower variation in the amount of bias across words. *Spacy* typically exhibited larger absolute mean values than *GloVe*, *word2vec*, and *SBERT*, except for its gender bias mean value, which was the lowest of all embeddings. *Spacy* also had larger standard deviations in the range of 8.6 to 14.8.

Finally, *BERT* and *GPT2* both exhibited some large mean values. For example, *BERT* had by far the largest absolute mean values for gender bias, age bias, and disability bias, while *GPT2* had the largest absolute mean value for ethnicity bias. This suggests that biases were encoded throughout word vectors in these embeddings. *BERT* also had the largest standard deviations, suggesting that its word vectors were widely dispersed around the bias directions. After debiasing, almost all of the mean bias scores became significantly smaller in absolute terms, with most very close to zero. The only major exceptions were *BERT* and *GPT2*, where mean bias scores remained high and actually increased in some cases.

Figure 7: Mean and standard deviation of Direct Bias across the embeddings before debiasing

	gender_mean	gender_std	ethnicity_mean	ethnicity_std	age_mean	age_std	region_mean	region_std	socioeconomic_mean	socioeconomic_std	disability_mean	disability_std
glove	3.96	9.040000	-1.15	7.69	-2.77	8.55	0.06	6.64	2.36	7.840000	-5.08	14.399999
w2v	0.87	7.040000	-0.01	7.54	-1.23	6.88	-0.67	5.82	1.65	7.160000	-2.10	7.350000
spacy	-0.47	8.610000	7.84	14.78	-6.87	9.48	-5.44	10.98	-0.14	10.640000	5.10	11.370000
sbert	-0.57	5.760000	1.40	6.48	-2.96	5.29	2.21	6.25	0.39	6.840000	5.80	6.790000
bert	7.83	28.760001	-2.96	16.62	8.46	21.03	5.65	11.55	-4.38	32.710001	11.05	24.169999
gpt2	-5.80	5.290000	8.58	5.12	-3.22	4.50	2.02	5.57	3.44	5.610000	-0.83	5.720000

Figure 8: Mean and standard deviation of Direct Bias across the embeddings after debiasing

	gender_mean	gender_std	ethnicity_mean	ethnicity_std	age_mean	age_std	region_mean	region_std	socioeconomic_mean	socioeconomic_std	disability_mean	disability_std
glove	0.0	0.0	-0.03	0.40	0.01	0.19	0.07	3.00	0.28	2.94	-0.10	0.50
w2v	-0.0	0.0	0.00	0.09	0.03	0.19	0.09	0.17	-1.04	4.31	0.23	0.49
spacy	-0.0	0.0	-0.72	1.01	-0.05	0.12	-0.47	0.96	1.94	2.59	-0.13	0.34
sbert	0.0	0.0	-0.02	0.12	0.03	0.18	-1.28	4.89	0.02	2.85	0.29	0.21
bert	-0.0	0.0	-4.43	1.79	4.62	1.38	-3.26	1.19	-7.58	2.79	5.87	1.58
gpt2	-0.0	0.0	3.48	0.29	1.46	0.20	-3.44	0.28	-1.58	0.25	4.39	0.23

It is important to recall that Direct Bias is only an effective bias measurement technique to the extent that the bias direction captures all information about the characteristic in the embedding. However, it is unlikely that this is the case in practise. One quick indication of this is to consider explained variance. Recall that the bias direction is found by taking the first principal component of the vector differences for definitional word pairs. The explained variance of the first principal component therefore tells us the amount of variation between these definitional word pairs that the bias direction accounts for. In our testing, none of the bias directions for any of the characteristics and embeddings except for *GPT2* explained more than 40% of the variance, meaning that the

bias directions failed to capture the majority of the information flow between the definitional word pairs.

Beyond this, there are also questions over the extent to which information about a characteristic is exhausted by the differences between definitional word pairs at all. As Gonen and Goldberg point out, biases are encoded *throughout* embeddings, not just in the bias direction. This is a major limitation of the Direct Bias measurement technique and limits the value of Hard Debiasing. While Hard Debiasing appears to have been successful in our testing because it reduced Direct Bias to close to zero in most cases, the process is *designed* to nullify Direct Bias, and is therefore only as effective as the bias direction is as a measurement of bias. The results in figure 8 should not be taken as a sign that Hard Debiasing has solved the problem.

Predicting gender association with KNN:

To demonstrate this flaw in Hard Debiasing, we followed Gonen and Goldberg's method of training a classifier to predict the prior Direct Bias association of the debiased word vectors. If a classifier has predictive power this would suggest that information about the bias was still present in the vectors after debiasing.

Taking the example of gender bias, we produced a binary 'gender bias' label for every vector based on whether its pre-debias Direct Bias score was greater than or less than zero. We then selected the 5000 most biased words in the embedding (2500 with each label) and randomly selected 1000 to constitute training data. We then trained a KNN model with k=5 to predict the 'gender bias' label from the debiased word vectors.

Hard Debiasing removes gender bias by neutralising the component of a vector that is aligned with the gender direction. If the gender direction fully captures the information about gender in the embedding, then the hard debiased vectors should hold no information about gender. Therefore, a model trained to predict the prior gender bias of the debiased vector should only have random accuracy. If the model's accuracy is greater than 50%, this would suggest that the vectors still contain information about the bias which is enabling the accurate predictions.

As it turned out, the KNN classifier was able to correctly identify the prior gender bias label of the 4000 debiased *GloVe* embeddings in the test set with 92% accuracy. We repeated this for every characteristic and embedding and found that accuracy was above 77% in every instance. See figure 9.

Figure 9: Accuracy scores for binary classification models trained to predict the prior bias association of debiased vectors

	gender	ethnicity	age	region	socioeconomic	disability
debiased_glove	0.92075	0.94575	0.95600	0.90725	0.96400	0.92300
debiased_w2v	0.88900	0.98000	0.95050	0.91475	0.97425	0.97100
debiased_spacy	0.94925	0.98900	0.96725	0.99425	0.98525	0.98325
debiased_sbert	0.89850	0.94050	0.94350	0.96650	0.97425	0.97100
debiased_bert	0.99925	0.99725	0.77975	0.87250	0.99950	0.77750
debiased_gpt2	0.99500	0.98500	0.99625	0.98850	0.99500	0.97125

This shows that information about the characteristics *is* still present in the word vectors after debiasing, which shows that debiasing failed to strip the vectors of bias and that

Direct Bias is not an exhaustive measure of bias. Even after removing the components that are aligned with the bias directions, the vectors are still clustered near vectors that share the same characteristic association.

5 Conclusion

Summary of results

Our research suggests that, at present, there is no single, definitive solution to the problem of bias in word embeddings.

First, no single bias metric provides a comprehensive view of bias. Our findings indicate that employing a variety of bias measurement techniques offers a more nuanced understanding of how bias manifests in embeddings. For instance, metrics like WEAT (Word Embedding Association Test) and Direct Bias provide different insights—while WEAT can reveal stereotypical associations between vectors, Direct Bias quantifies how much information about characteristics like gender or ethnicity is encoded in each vector.

The combined use of multiple bias measurement techniques yielded more insightful results than relying on any single metric. For example, WEAT was effective at identifying gendered stereotypes, such as older people being more closely associated with responsibility than younger people. In contrast, Direct Bias helped quantify the extent to which specific vectors encode demographic information. While neither metric captured bias in its entirety, the comparison of embeddings using these metrics revealed differences in how bias was encoded. For instance, comparing two embeddings could show that one had a lower Direct Gender Bias, suggesting it may be more suitable for certain applications.

Second, even when multiple metrics are considered together, the inherent limitations of existing measurement and mitigation techniques indicate that addressing bias in embeddings remains a complex task. Evidence points to the need for context-specific and systematic evaluations of bias in NLP applications that use embeddings. This reflects the reality that bias is highly context-dependent and embedded in a wide range of linguistic and social factors.

Third, post-processing techniques, such as Hard Debiasing, proved in our research to be unreliable in completely removing bias. Although these methods can reduce certain bias metrics, they often fail to address bias comprehensively, as seen in our analogy tests where debiasing compromised the overall accuracy of embeddings. These findings align with the broader literature, which suggests that while debiasing techniques have some utility, they may not fully eliminate bias.

Further Research

Future academic research in this area could also focus on bias measurement and mitigation techniques designed specifically for contextual and sentence embedding models. Our research showed that techniques designed for ‘static’ embeddings may not well suited to these more advanced embedding models, which are now finding wide use in Retrieval Augmented Generation (RAG)-style applications.

Further academic research could also focus on testing for bias in final products, for example, examining whether a chatbot's responses differ when queries are phrased in

various linguistic styles or incorporate different demographic information. Additionally, understanding whether biases present in embeddings correlate with biases in the outputs of such applications could be an important avenue for future research. While our study did not focus on this aspect, it stands as a promising direction for further inquiry.

Finally, engaging consumers and end-users in research efforts could also provide meaningful insights into how bias impacts their experiences with these technologies.

Annex 1: Data

Hard debias

Table 4: Definitional word pairs used to define the bias directions.

Characteristic	Definitional word pairs
Gender	['woman', 'man'], ['girl', 'boy'], ['she', 'he'], ['mother', 'father'], ['sister', 'brother'], ['daughter', 'son'], ['gal', 'guy'], ['female', 'male'], ['hers', 'his'], ['her', 'him'], ['herself', 'himself'], ['wife', 'husband'], ['mum', 'dad'], ['uncle', 'aunt'], ['grandmother', 'grandfather'], ['granddaughter', 'grandson'], ['niece', 'nephew'], ['queen', 'king'], ['princess', 'prince'], ['madam', 'sir'], ['lady', 'lord'], ['ladies', 'gentlemen'], ['bride', 'groom'], ['mrs', 'mr'], ['bachelorette', 'bachelor'], ['lass', 'lad'], ['girlfriend', 'boyfriend'], ['stepmother', 'stepfather'], ['feminine', 'masculine'], ['heiress', 'heir'], ['duchess', 'duke'], ['baroness', 'baron'], ['empress', 'emperor'], ['governess', 'governor'], ['motherhood', 'fatherhood'], ['matriarchy', 'patriarchy'], ['jill', 'jack'], ['eve', 'adam'], ['sisterhood', 'brotherhood'], ['sorority', 'fraternity'], ['womanhood', 'manhood'], ['actress', 'actor']
Age	['old', 'young'], ['older', 'younger'], ['age', 'youth'], ['elderly', 'youthful'], ['mature', 'immature'], ['elder', 'youngster'], ['senior', 'junior'], ['adult', 'child'], ['boomer', 'millennial']
Ethnicity	['migrant', 'citizen'], ['minority', 'majority'], ['immigrant', 'native'], ['foreign', 'english'], ['eastern', 'western'], ['asia', 'europe'], ['asian', 'european'], ['developing', 'developed'], ['south', 'north'], ['brown', 'white']
Disability	['disabled', 'able'], ['disability', 'ability'], ['dependent', 'independent'], ['neurodiverse', 'neurotypical'], ['sickness', 'health'], ['handicap', 'advantage'], ['weakness', 'strength'], ['impaired', 'unimpaired'], ['immobile', 'mobile'], ['unwell', 'well'], ['unhealthy', 'healthy'], ['unfit', 'fit']
Socio-economic background	['rich', 'poor'], ['wealthy', 'unwealthy'], ['privileged', 'unprivileged'], ['professional', 'labourer'], ['homeowner', 'renter'], ['advantaged', 'disadvantaged'], ['work', 'unemployed'], ['salary', 'wages'], ['affluent', 'impoverished'], ['prosperous', 'deprived'], ['elite', 'marginalised'], ['bourgeois', 'proletariat'], ['manager', 'worker'], ['aristocracy', 'commoner'], ['employment', 'benefits'], ['secure', 'insecure']
Region	['north', 'south'], ['northern', 'southern'], ['northerner', 'southerner'], ['scotland', 'england'], ['industrial', 'market'],

	['agricultural', 'rural'], ['manchester', ' london'], ['lancashire', 'surrey'], ['yorkshire', 'kent'], ['liverpool', 'bristol'], ['glasgow', 'edinburgh'], ['industry', 'services']

Word analogy test

Analogies were generated from sets of characteristic word pairs and attribute and financial words. The characteristic word pairs were:

- ['man','woman'],['white','black'],['old','young'],['able','disabled'],['london','liverpool'],['rich','poor']

And the attribute words were:

- 'educated', 'successful', 'rich', 'senior', 'employed', 'responsible', 'innovative', 'health', 'finance', 'mortgage', 'loan', 'credit', 'insurance', 'invest', 'bank', 'save'

Analogies were formed with the format "characteristic_word_1 is to attribute_word as chacateristic_word_2 is to …". For example, "man is to educated as woman is to …".

Table 5: Number of incorrect analogies produced by each embedding.

Embedding	Number of false analogies before debasing	Number of false analogies after debiasing
GloVe	33	66
word2vec	22	51
Spacy	26	53
SBERT	4	35
BERT	53	87
GPT2	88	90

Word Embedding Association Test

The WEAT relies on sets of target and attribute pairs to be defined and calculates the differential similarity between those sets. The target sets represent demographic characteristics and can be seen in table 4, where they are presented as definitional word-pairs. The attribute sets can be seen below in table 6. For each attribute we defined two set of words that characterise both dimensions of the attribute. For example, for the attribute *education*, one set included the words 'educated,' 'literate,' and 'graduate,' while the other set included the words 'uneducated,' 'illiterate,' and 'dropout.'

Table 6: Attribute sets and the words used to define them.

Embedding	Definitional words
Work and family	'work', 'job', 'salary', 'earn', 'career', 'commute', 'employment', 'occupation', 'professional', 'office', 'business', 'promotion', 'full-time', 'primary earner'
	'home', 'family', 'children', 'care', 'house', 'parent', 'marriage', 'domestic', 'household', 'part-time'
Seniority	'manager', 'senior', 'executive', 'experienced', 'skilled', 'director', 'leader', 'expert', 'partner', 'tenure', 'status', 'successful'
	'junior', 'inexperienced', 'unskilled', 'assistant', 'associate', 'trainee', 'intern', 'entry-level', 'support', 'aide', 'help', 'temporary'
Risk	'risk seeking', 'high risk tolerance', 'risk taking', 'speculative', 'volatility', 'long-term', 'entrepreneurial', 'confident'
	'cautious', 'risk averse', 'low risk tolerance', 'conservative', 'careful', 'prudent', 'safety', 'security', 'protective'
Responsibility	'secure', 'ambitious', 'responsible', 'responsibility', 'security', 'safe', 'safety', 'reliable', 'reliability', 'dependable', 'mature', 'predictable', 'established', 'steady', 'consistent', 'career-oriented', 'prosperous', 'motivated', 'wealth', 'professional', 'organized', 'experience', 'settled'
	'unstable', 'irresponsible', 'thoughtless', 'insecure', 'unreliable', 'undependable', 'immature', 'inconsistent', 'inexperienced', 'careless', 'impulsive', 'unpredictable', 'unsteady', 'careerless', 'reckless', 'aimless', 'unambitious', 'unmotivated', 'volatile', 'turbulent', 'disorganized', 'chaotic'
Health	'health', 'energy', 'energetic', 'enthusiastic', 'fit', 'healthy', 'vitality', 'vibrant', 'active', 'agile', 'lively', 'robust', 'strong', 'athletic', 'spirited', 'well', 'flexible', 'endurance', 'stamina'
	'unhealthy', 'weak', 'inactive', 'fatigued', 'lazy', 'frail', 'feeble', 'sick', 'ill', 'suffer', 'illness', 'disability', 'unfit', 'limp', 'unwell'
Income	'employed', 'high income', 'high-earning', 'lucrative', 'well-paid', 'high salary', 'successful', 'prosperous'
	'unemployed', 'low income', 'low salary', 'underpaid', 'unpaid', 'uncompensated', 'struggling'
Wealth	'owner', 'homeowner', 'landowner', 'landlord', 'save', 'savings', 'saver', 'asset', 'assets', 'asset holder', 'invest', 'investment', 'investor', 'property', 'wealth', 'rich', 'posh', 'wealthy', 'affluent', 'properous', 'advantaged', 'ownership', 'opulent', 'aristocratic', 'upper class', 'middle class', 'elite', 'privileged', 'expensive', 'luxurious', 'prestigious', 'luxury', 'prestige',

	'sophisticated', 'extragant', 'noble', 'comfortable', 'well-to-do', 'moneyed', 'estate', 'endowment', 'endowed', 'inheritance'
	'rent', 'renter', 'council house', 'social house', 'debt', 'debtor', 'borrower', 'liability', 'poor', 'common', 'working class', 'lower class', 'disadvantaged', 'impoverished', 'needy', 'blue-collar', 'struggling', 'frugal', 'cheap', 'destitute', 'impoverished', 'poverty', 'deprived', 'underprivileged', 'insecure', 'hard-up'
Education	'education', 'educated', 'informed', 'literate', 'university', 'bachelors', 'masters', 'doctorate', 'graduate', 'academic', 'research', 'wise', 'wisdom', 'scholar', 'postgraduate', 'thesis', 'dissertation', 'intellectual', 'learned', 'expertise', 'knowledge', 'skill', 'proficiency'
	'uneducated', 'illiterate', 'uninformed', 'dropout', 'unskilled', 'ignorant', 'naive', 'unlearned', 'basic', 'beginner', 'amateur', 'untaught', 'unwise'

Table 7: WEAT effect sizes and p-values.

Table showing the WEAT effect sizes and p-values for 10 hypothesis tests across 6 pre- and post-debiased embeddings. P-values in brackets. Column headers refer to an individual hypothesis – see key below.

Embedding	1	2	3	4	5	6	7	8	9	10
GloVe	-0.529 (0.867)	0.665 (0.092)	-1.75 (1.00)	0.372 (0.202)	1.055 (0.008)	-0.813 (1.00)	-0.55 (0.995)	0.187 (0.331)	0.874 (0.013)	1.487 (0.00)
Debiased GloVe	-0.057 (0.536)	0.638 (0.097)	-1.95 (1.00)	0.095 (0.413)	-0.012 (0.516)	0.012 (0.473)	-0.035 (0.567)	-0.207 (0.686)	0.171 (0.315)	0.764 (0.021)
word2vec	-0.512 (0.852)	0.825 (0.046)	-1.77 (1.00)	0.055 (0.451)	0.768 (0.042)	-0.5 (0.987)	-0.57 (0.996)	0.139 (0.368)	1.063 (0.004)	1.383 (0.00)
Debiased word2vec	-0.059 (0.539)	0.763 (0.052)	-1.96 (1.00)	0.082 (0.422)	0.037 (0.458)	-0.031 (0.55)	-0.006 (0.508)	0.14 (0.372)	-0.074 (0.574)	0.793 (0.019)
Spacy	-0.844 (0.957)	0.309 (0.257)	01.58 (1.00)	-0.17 (0.65)	-0.223 (0.696)	-0.315 (0.921)	-0.49 (0.991)	0.332 (0.217)	0.615 (0.054)	1.536 (0.00)
Debiased Spacy	-0.366 (0.777)	-0.587 (0.881)	-1.84 (1.00)	0.081 (0.423)	0.001 (0.51)	-0.04 (0.58)	0.001 (0.495)	0.001 (0.496)	0.016 (0.48)	1.324 (0.00)
SBERT	-0.702 (0.923)	0.911 (0.032)	-1.71 (1.00)	-0.003 (0.504)	0.179 (0.345)	-0.535 (0.994)	-0.269 (0.89)	0.592 (0.083)	1.057 (0.003)	1.46 (0.00)
Debiased SBERT	0.168 (0.369)	1.107 (0.013)	-1.64 (1.00)	-0.217 (0.678)	-0.003 (0.497)	-0.007 (0.52)	0.01 (0.477)	-0.208 (0.69)	0.21 (0.301)	0.619 (0.05)
BERT	-0.319 (0.735)	-0.007 (0.513)	-1.17 (0.997)	0.193 (0.326)	-0.419 (0.831)	-0.38 (0.959)	0.09 (0.35)	-0.069 (0.57)	0.585 (0.064)	1.022 (0.004)
Debiased BERT	-0.017 (0.506)	0.679 (0.078)	-0.464 (0.852)	0.045 (0.456)	0.012 (0.492)	-0.043 (0.582)	0.008 (0.489)	0.114 (0.393)	0.038 (0.46)	0.634 (0.051)

GPT2	0.015 (0.484)	0.074 (0.438)	-1.03 (0.991)	0.894 (0.02)	0.505 (0.135)	-0.025 (0.544)	-0.031 (0.544)	-0.249 (0.711)	0.564 (0.074)	0.603 (0.061)
Debiased GPT2	0.157 (0.375)	-0.848 (0.956)	-0.635 (0.932)	0.018 (0.488)	-0.036 (0.534)	-0.022 (0.533)	-0.032 (0.573)	0.143 (0.369)	0.164 (0.342)	0.048 (0.459)

Key:

1. Age with respect to health
2. Age with respect to responsibility
3. Disability with respect to health
4. Disability with respect to risk
5. Ethnicity with respect to risk
6. Gender with respect to seniority
7. Gender with respect to work and family
8. Region with respect to income
9. Socioeconomic background with respect to education
10. Socioeconomic background with respect to wealth

Table 8: The most biased words for each characteristic and embedding.

For each characteristic and embedding we show the top 6 most biased words in each direction. For example, for gender, we show the six most male and the six most female words in each embedding.

Embedding	Gender	Ethnicity	Age	Region	Socioeconomic background	Disability
GloVe	(succeeded, colonel, john, chairman, George, general) (lactating, songstress, barmaid, ditzy, needlework, comedienne)	(name, heraldry, inventor, Scottish, distinguished, invented) (laborers, migrants, unskilled, remittances, inflows, farmworkers)	(physicality, athleticism, exuberant, underachieving, rejuvenated, scrappy) (died, wife, families, care, married, surviving)	(medi, meads, hawk, aol, qwest, montomgery) (textile, machinery, metallurgy, smokestacks, metalworking, unido)	(elegant, luxurious, boutique, ken, frank, baroque) (chronically, handicapped, malnourished, undernourished, destitute, disenfranchised)	(better, way, could, own, come, bring) (bedridden, chastises, nauseous, malnourishment, nauseated, vomited)
word2vec	(boyhood, countryman, jnr, journeyman, beard, patriarch) (songstress, heroine, chanteuse, comedienne, businesswoman, housewife)	(oma, nc, Lowell, shur, resides, crawford) (migrants, immigrants, boatpeople, laborers, farmworkers, immigration)	(talent, youngsters, starlet, positivity, dynamism, talents) (homebound, inheritance, disabled, caregiver,	(abutting, janitorial, custodial, habilitation, recreation, residential) (marketplace, mkt, markets, industrywide,	(advantages, platform, leverage, sophisticated, footprint, leveraging) (homeless, destitute, jobless, malnourished,	(positioned, opportunity, strengths, leverage, enabled, execute) (bedridden, disabilities, paraplegic,

			medicaid, apartment)	oversupply, profitability)	illiterate, malnourishment)	quadriplegic, pensioner, sick)
Spacy	(kinsman, usurper, kingpin, conqueror, renegade, abbot) (seductress, lolita, brunette, xo, lingerie, catwman)	(governments, nongovernment, government, foreigners, diplomats, antigovernment) (urdu, Telugu, hindi, kore, tagalog, iit)	(talent, passionate, enthusiasm, enthusiasms, inspire, inspires) (inch, inchoate, died, percent, cm, km)	(staffordshire, berkshire, cricketing, tottenham, noncash, shire) (russo, Casanova, pistol, klein, corky, ron)	(stylish, vibrant, blend, versatile, elegant, boutique) (eta, afflicted, sicko, inadequacy, absenteeism, abused)	(effortlessly, customize, versatility, customizable, flexibility, combo) (ill, niu, eta, sickbed, sick, sicko)
SBERT	(hes, sons, mans, brothers, mang, lads) (sportswoman, Sophia, females, actresses, teresa, hilary)	(eu, euro8, euro, euro1, euro3, euros) (asians, sarong, zhou, asean, sahitya, oriental)	(youngs, youngsters, children, childlike, youths, adolescents) (seniors, elders, geriatric, emeritus, geriatrics, obituary)	(southerners, souths, scs, Charleston, memphis, sc) (nitride, nni, nord, northerly, northernmost, northland)	(advantage, advantageous, advantages, beneficial, benefit, benefitted) (joblessness, deprivations, lawlessness, deprivation, homeless, orphans)	(stronger, bigger, tighter, abilities, bout, gained) (disabilities, absenteeism, impairments, wheelchairs, incapacitated, absentia)
BERT	(popstar, condominiums, occultism, lacoste, jeweler, stepson) (absent, grounded, sac, failing, puerto, fame)	(eurozone, euro1, cri, signatory, cooperated, popstart) (oriental, prostate, hangul, Indonesian, sutra, Astro)	(youngsters, grownups, toddlers, toddler, childlike, boyish) (tori, relocating, opaque, heartland, skinned, ebony)	(travelogues, travelogue, lms, clicker, sender, dsp) (industries, economic, textile, physics, medical, manufacturing)	(chastises, chastising, chastised, autoimmune, fawning, prolapse) (blackout, tubular, publisized, relocating, orbits, grounded)	(daze, cuff, opaque, heartland, cowboys, reforming) (unfunded, unsecured, unrepresented, unattended, unsteady, uninhabitable)
GPT2	(fortunately, quartered, ultimate, pointers, looking, amines) (excuses, excavator, excursions, excreted, excavating, excursion)	(fortunately, quartered, pointers, ultimate, amines, looking) (excavator, excusing, excuses, excavations, excavation, excursion)	(angering, governmental, dominated, quartered, inducing, classifieds) (psc, excoriated, excuses, excused, excavated, excavators)	(quartered, fortunately, ultimate, looking, angering, pointers) (excused, excuses, excavator, excusing, excitatory, exciting)	(quartered, fortunately, looking, ultimate, angering, pointers) (excursion, exciting, excoriated, excavating, excitatory, excavators)	(quartered, fortunately, angering, looking, ultimate, pointers) (excavators, excused, excites, excited, excuse, excitement)

Annex 2: References

Bughin, J., Seong, J., Manyika, J., Chui, M., Joshi, R. (2018). *Notes from the AI frontier: Modelling the impact of AI on the world economy*. McKinset & Company. https://www.mckinsey.com/~/media/McKinsey/Featured%20Insights/Artificial%20Intelligence/Notes%20from%20the%20frontier%20Modeling%20the%20impact%20of%20AI%20on%20the%20world%20economy/MGI-Notes-from-the-AI-frontier-Modeling-the-impact-of-AI-on-the-world-economy-September-2018.ashx

Dennehy, F. (2024). *Large Language Models could revolutionise finance sector within two years*. The Alan Turing Institute. https://www.turing.ac.uk/news/large-language-models-could-revolutionise-finance-sector-within-two-years

O'Neill, M. (2023, October 23). What is robo-advice and how can it provide low-cost financial planning? *Financial Times*. https://www.ft.com/content/6694bb4a-a585-496a-b7f3-d1841984f9b3

Lumley, L. (2023). Large language models advance on financial services. *The Banker*. https://www.thebanker.com/Large-language-models-advance-on-financial-services-1678359788

Goller, C. (2023, March 30). Word Embeddings: a technology still relevant?. *INTRAFIND*. https://intrafind.com/en/blog/word-embeddings-technology-still-relevant

Financial Conduct Authority. (2022). *A new Consumer Duty (PS22/9)*. https://www.fca.org.uk/publication/policy/ps22-9.pdf

Gibney, E. (2024, March 13). Chatbot AI makes racist judgements on biases of dialect. *Nature*. https://www.nature.com/articles/d41586-024-00779-1

Office for Artificial Intelligence. (August, 2023). *A pro-innovation approach to AI regulation*. Department for Science, Technology & Innovation. https://www.gov.uk/government/publications/ai-regulation-a-pro-innovation-approach/white-paper

Financial Conduct Authority. (April, 2024). *Artificial Intelligence (AI) update - further to the Government's response to the AI White Paper*. https://www.fca.org.uk/publications/corporate-documents/artificial-intelligence-ai-update-further-governments-response-ai-white-paper

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR*. https://doi.org/10.48550/arXiv.1301.3781

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of NIPS*. https://doi.org/10.48550/arXiv.1310.4546

Mikolov, T., Yih, W., Zweig, G. (2013). Linguistic Regularities in Continuous Space Word Representations. *Proceedings of NAACL HLT*. https://aclanthology.org/N13-1090

Barla, N. (2024, April 16). The Ultimate Guide to Word Embeddings. *neptune.ai*. https://neptune.ai/blog/word-embeddings-guide

Bender, E., McMillan-Major, A., Shmitchell, S., & Gebru, T. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? *FAccT '21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–623. https://doi.org/10.1145/3442188.3445922

Dastin, J. (2018, October 11). Insight - Amazon scraps secret AI recruiting tool that showed bias against women. *Reuters*. https://www.reuters.com/article/world/insight-amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK0AG/

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. https://doi.org/10.3115/v1/d14-1162

University of Edinburgh. (March, 2024). King - man + woman = queen: the hidden alegbraic structure of words. https://informatics.ed.ac.uk/news-events/news/news-archive/king-man-woman-queen-the-hidden-algebraic-struct

Caliskan, A., Bryson, J. J., & Narayanan, A. (2016). Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334), 183–186. https://doi.org/10.1126/science.aal4230

Bolukbasi, T.; Chang, K.; Zou, J. Y.; Saligrama, V.; and Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *ArXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1607.06520

Mukul Rathi. (2021, February 28). *Does debiasing word embeddings actually work? (+ explanation of GN-GloVe, Hard-debiasing)* [Video]. YouTube. https://www.youtube.com/watch?v=MKs2io7opJs

Gonen, H., & Goldberg, Y. (2019). Lipstick on a Pig. *Proceedings of the 2019 Conference of the North*. https://doi.org/10.18653/v1/n19-1061

Zhao, J., Zhou, Y., Li, Z., Wang, W., & Chang, K.-W. (2018). Learning Gender-Neutral Word Embeddings. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1809.01496

Caliskan, A., Ajay, P. P., Charlesworth, T., Wolfe, R., & Banaji, M. R. (2022). Gender Bias in Word Embeddings. *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. https://doi.org/10.1145/3514094.3534162

Zhang, B. H., Lemoine, B., & Mitchell, M. (2018). Mitigating Unwanted Biases with Adversarial Learning. *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. https://doi.org/10.1145/3278721.3278779

Brunet, M.-E., Alkalay-Houlihan, C., Anderson, A., & Zemel, R. (2018). Understanding the Origins of Bias in Word Embeddings. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1810.03611

Explosion AI. (2020). spaCy: Industrial-strength Natural Language Processing in Python (Version 2.2.4) [Software]. https://spacy.io/

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1, 4171-4186. ttps://arxiv.org/abs/1810.04805

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI*. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

Reimers, N., & Iryna Gurevych. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. https://doi.org/10.48550/arxiv.1908.10084

Badilla, P., Bravo-Marquez, F., & Perez, J. (2020). WEFE: The Word Embeddings Fairness Evaluation Framework. *Proceedings of the 29th International Joint Conference on Artificial Intelligence and the 17th Pacific Rim International Conference on Artificial Intelligence (IJCAI-PRICAI 2020), Yokohama, Japan*. https://doi.org/10.24963/ijcai.2020/60

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.

12 Endeavour Square, London E20 1JN
Telephone: +44 (0)20 7066 1000 Website: www.fca.org.uk